THE DETERMINATION OF
EMPIRICAL AND ANALYTICAL
SPACECRAFT PARAMETRIC CURVES
- THEORY AND METHODS -

FINAL PROGRESS REPORT

Submitted

by the

TEXAS A&M RESEARCH FOUNDATION

to the

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

for NASA Grant

SC-NGR-44-001-027

Prepared

by

THE INDUSTRIAL ENGINEERING DEPARTMENT

of

TEXAS A&M UNIVERSITY

May 1, 1967

FOREWORD

This document represents the final progress report on the NASA research grant NGR 44-001-027. The report is divided into three parts. A summary of each section is presented below.

Part I is a rather thorough treatment of an algorithm which was developed to assist in the development of cost estimating relationships. The general application is to permit a minimum sum of squares approach to fitting a cost estimating relationship, based upon the constraints of minimizing or maximizing the variable costs in a hardware development program with the coefficients being restricted to non-negative values.

Part II is an extension of the run-out cost estimation problem with generalized constraints placed upon the least-squares estimation of the polynomial being used to represent percent cost-percent time of the program. The technique developed uses a weak constraint of non-negative slopes on the tangent to the cumulative cost curve. This procedure provides the minimum least squares possible under this constraint.

Part III is a new area of development in the cost research grant in that it is directed toward relating hardward deliveries to cost and the segregation of variable and non-variable costs.

CONVEX PROGRAMMING APPLIED TO THE ESTIMATION OF THE PARAMETERS OF DEFINITE QUADRATIC FORMS AND TO RELATED TESTS OF HYPOTHESES

by

William P. Cooke, Jr.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

A. The Problem

Response surface analysis in operational research is concerned with the relationship of a 'response', y, and a number of 'inputs' $x_1, x_2, \ldots, x_k$. Often this relationship can be approximated by a mathematical equation which is a so-called second order polynomial in the x_i. Such a polynomial involves only terms of the form x_i, x_i^2, and $x_i x_j$. The basic task in response surface analysis is to determine the unknown coefficients of this second order response surface, using pilot data in which for a number of 'experiments' associated inputs x_i and outputs y have been recorded.

The customary technique of estimating these coefficients is by 'least squares'. Frequently, however, additional information about the response surface is available. In this dissertation techniques will be developed which modify the above least squares procedure so that such additional information, of a specific type, can be utilized. The utility of the procedure when applied to a test of the hypothesis that the response surface is of a special type will also be demonstrated.

Since the least squares procedure when applied to a linear model requires the minimization of a certain quadratic form, a general procedure for minimizing a quadratic form subject to certain restrictions is required. The specific problem to be considered follows.

Consider a set of N responses y_t, $t = 1, 2, \ldots, N$ and associated input vectors of the form $X_t' = (x_{1t}, x_{2t}, \ldots, x_{kt})$ and assume that the expected response $E(y_t)$ is a second order function of the inputs x_{it} with unknown coefficients. More specifically, assume the model

$$E(y_t) = \beta_{00} + \beta_1' X_t + X_t' BX_t \tag{1}$$

where $\beta_1' = (\beta_{10}, \ldots, \beta_{k0})$ and $B = (\beta_{ij})$, $i,j = 1, \ldots, k$. Further, assume that $y_t - E(y_t) = e_t$ where the e_t are independent, normal variates with mean zero and variance σ^2.

Suppose now that it is known that the matrix B is positive semi-definite (or negative semi-definite). Such situations frequently occur in the final stages of response surface analysis, see Davies (1956), or again in 'production economics', see Heady and Dillon(1961), when it is known that a model for a multiple-input production function is meaningless if it has a saddle-point.

The problem then is to estimate the unknown parameters β_{ij}, $i,j = 0, 1, \ldots, k$ subject to the restriction that the matrix B is positive (or negative) semi-definite. The least squares principle is to be used, so that an equivalent statement of the problem is to minimize, as a function of β_{ij}, the quadratic form

$$Q(\beta) = \sum_{t=1}^{N} (y_t - E(y_t))^2 \tag{2}$$

subject to the restriction that $E(y_t)$ is a positive (or negative) semi-definite quadratic form. This problem is solved in Chapter II.

In addition, once a procedure for estimating the β_{ij} is established, a related problem will be considered. Suppose that instead of knowing that B is a semi-definite matrix it is desired to test the hypothesis that B is semi-definite. That is, it will be of value in response surface analysis to have the capability of testing the hypothesis that the surface is of a type that does not have a saddle-point or, even more importantly, that the surface possesses a unique minimum or maximum. Procedures for testing such hypotheses are discussed in Chapters III and IV.

The solution of both the problems described above will employ a convex programming algorithm developed by Hartley and Hocking (1963). A brief description of the algorithm as applied to the problem at hand is found in Chapter II.

B. Historical Background

Before considering the general question of estimation of parameters under constraints it is of interest to review the method of least squares as applied to the problem of unconstrained estimation. Consider then the general linear model

$$Y = X\beta + e \tag{3}$$

where Y is an Nx1 vector of observations, X is an Nxn matrix of known constants, β is an nx1 vector of unknown constants or parameters which are to be estimated, and e is an Nx1 vector of errors with the property thet $e \sim MVN(0, \sigma^2 I)$.

Under these conditions the least squares procedure will be equivalent to the method of maximum likelihood. The best linear unbiased estimate of the unknown vector β is obtained by minimizing

$$Q(\beta) = e'e = (Y - X\beta)'(Y - X\beta). \quad (4)$$

The vector of estimates is

$$\hat{\beta} = (X'X)^{-1}X'Y \quad , \quad (5)$$

and the properties of these estimates are well-known, see Graybill (1961).

If now there are restrictions imposed on the parameter vector β in the form of a set of p linear equations, where $p \leq n$, the least squares solution can be obtained by the method in (4) and (5) after a simple linear transformation. The properties of these estimates are also known (Graybill(1961)).

The two problems mentioned above might be called the 'classical' least squares problems, whose solutions have been known since before 1900. If now we regard the parameter vector β to be restricted to a convex subspace of n-dimensional Euclidean space, E_n, the problem takes on a more 'modern' aspect. If the problem were to minimize a linear function subject to linear inequalities, we have what is known as a linear programming problem. A general method for solution of this problem, called the Simplex method, has been available since 1958, see Dantzig (1948).

More generally the problem of finding the extrema of functions subject to convex restrictions is called a mathematical programming problem. A good review of current methods and results in that area may be found in Dantzig (1963) and Graves and Wolfe (1963).

The particular problem of this paper deals with a quadratic objective function, the function which is to be minimized, subject to the restriction that β lies in a convex subspace of E_n. Now if the subspace, S, can be specified by a finite set of linear inequalities the problem would be called a quadratic programming problem. Various solutions to such a problem have been developed, examples being those of Beale (1955) and Wolfe (1959). Since standard quadratic programming techniques will be found inapplicable to the problem at hand, they will not be discussed in detail here.

A particular application to a statistical problem of this type can be found in Lewish (1963). While the problems considered in the Lewish paper are in some ways similar to those considered in this dissertation, and in fact some of Lewish's results apply directly to the current problem, Lewish was only considering problems to which known quadratic programming techniques could be applied. The large contribution of Lewish was to determine the statistical properties of the estimates so obtained, an area of research that had been largely ignored by workers in the field of mathematical programming.

It will be shown in Chapter II that while the restriction space S is convex for our particular problem it cannot be specified by a finite set of linear inequalities. Thus, while the objective function is quadratic, some technique other than quadratic programming must be used.

While the specific estimation problem of this dissertation has received little attention in available literature, the general area of response surface analysis has enjoyed more popularity, especially since 1951.

An early paper in the same vein as what is now known as response surface analysis is that of Rice (1939) in which an expression is derived for the probability that a random function, the parameters being random with known distribution, of a single variable possesses a maximum in some small rectangular region. While subsequent research on response surfaces has followed a different path, it will be seen that the discussion in Chapter IV of this paper bears some resemblance, in a multivariate sense, to Rice's original idea.

The article more generally regarded as being among the first to broach the question of the experimental determination of optimum conditions is the paper by Hotelling (1941). Hotelling contributed the questions answered by Box and Wilson (1951) in their classic paper, namely those of how to approach a stationary point and how to find it once in its neighborhood. Here the estimation of parameters was firmly established as the basic operation in response surface analysis.

Aitchison and Silvey (1958) discussed the asymptotic distribution of a 'restricted maximum liklihood estimator' as well as a test of the hypothesis that the true parameter lies in the subset specified by the linear restriction. Theil (1963) considered the question of prior information in a regression context. He also considered a test of the hypothesis that prior and sample information are in agreement with each other. It will be noted that the preceding two papers included tests of hypotheses of a type that we will be considering. However, neither affords a test for the specific hypothesis that will be tested in this dissertation.

A recent paper by Judge and Takayama (1966) applies quadratic programming to regression problems with various specified linear inequality restrictions. Judge and Takayama apparently have solved such problems for a wide variety of possible restrictions but the case of infinitely many linear restrictions, as we have here in our problem, is not amenable to solution by their methods.

In the convex programming algorithm of Hartley and Hocking (1963) is found the means of solution for the estimation problem, and as will be made apparent, the hypothesis test problem as well. Since the Hartley-Hocking algorithm requires that the constraints be specified by convex functions, it will be made clear in Chapter II why an infinity of linear restrictions are specified rather than a simpler description of S which does not consist wholly of convex functions.

Another paper warranting mention as an illustration of a situation where the experimenter may well have used the results of Chapter II is that of Tramel (1963). Tramel writes of an experiment conducted by Mississippi State University scientists to determine the economically optimum levels of three chemical fertilizers for cotton. Twenty-six 'production functions' were fit by standard least squares with the result that fourteen of the twenty-six functions had "illogical signs" for some of the parameters. The precise difficulty was that some of the second-degree terms involving a single variable had negative coefficients. The conclusion reached was,

> "...the usefulness of continuous functions as a means of estimating response surfaces in cotton fertility experiments is questionable. ...Form-free estimation of points on the response surface would appear to be the preferred alternative."

It would seem that in the experiment described above there was some reason to begin with the assumption that a good approximation to the actual production function would be a continuous function, else there would have been no attempt to estimate its parameters. The original assumption apparently was abandoned not because it was wrong but because it was impossible to obtain parameter estimates compatible with the prior knowledge that the production function should be a semi-definite quadratic form in the input variables.

Apparently the specific hypotheses test we will make has not been discussed in available literature. Probably this is because the estimation problems required had not been solved. Hopefully, now that the problem of parameter estimation is solved and a test procedure for the hypothesis has been proposed, experimenters will want to both use and improve upon these initial results.

CHAPTER II

LEAST SQUARES FIT OF DEFINITE QUADRATIC FORMS

A. Description of the Problem

The model for the estimation problem was described in section I(1). Suppose now that is is known that the matrix B of model I(1) is positive (or negative) semi-definite. Since the results to be derived apply to either case with only minor differences in formulation we will henceforth suppose only that B is positive semi-definite.

The problem is to estimate the β_{ij}, i, j = 0, 1, ... , k, subject to the above restriction, in such a fashion that the estimates have desirable statistical properties.

A procedure that will be shown to lead to such estimates is that of 'restricted least squares'. Specifically, the method will be to find the vector β^* which minimizes the quadratic

$$Q(\beta) = \sum_{t=1}^{N} (y_t - E(y_t))^2 \tag{1}$$

subject to the condition that $B^* = (\beta^*_{ij})$, i,j = 1, ... , k is positive semi-definite, where β is the vector of all unknown parameters in $E(y_t)$.

In section II.B it is shown that the requirement that the matrix B is semi-definite restricts the β_{ij}, i,j = 1, ... , k to a convex subset, say S, of the $\binom{k+1}{2}$-dimensional β-space and hence the estimation of the β_{ij} by minimization of the quadratic (1) subject to this restriction is a convex programming problem.

The particular specification of the subset S will be of great importance to the practicability of solution of the problem and merits some discussion. Perhaps the more familiar mode of specification of S is the set of inequalities arising from the condition that all principal minors of the matrix B have non-negative determinants. Such specification does result in a finite number of inequality restrictions on functions of the β_{ij}. However, although the region S defined by these inequalities is a convex region the functions defined by the determinants are not, in general, convex. Thus we have the rather unusual situation of a convex region being specified by functions which are not necessarily convex functions. A simple example is presented below to illustrate this situation.

Consider the positive definite matrix

$$A = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \tag{2}$$

where $a_1 > 0$, $a_4 > 0$, $a_1a_4 - a_2a_3 \quad 0$. Let $f_1(A) = a_1$, $f_2(A) = a_4$, and $f_3(A) = a_1a_4 - a_2a_3$. Now the definition of a convex function f over a set S requires that, for any two points P_1, P_2 in S,

$$f(\lambda P_1 + (1 - \lambda)P_2) \leq \lambda f(P_1) + (1 - \lambda)f(P_2) \tag{3}$$

for all λ such that $0 \leq \lambda \leq 1$. It is easily verified that $f_1 > 0$ and $f_2 > 0$ are in fact convex functions. We will now show that $f_3 > 0$ is not convex.

Let

$$A_1 = \begin{pmatrix} 1 & 2 \\ 2 & 5 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & 3 \\ 3 & 10 \end{pmatrix} \tag{4}$$

so that the elements of A_1 and A_2 are seen to lie in S; that is, A_1 and A_2 are positive definite. Then

$$f_3(A_1) = 1 \ , \ f_3(A_2) = 1 \quad . \tag{5}$$

Let $\lambda = 1/2$. Then

$$\lambda A_1 + (1 - \lambda)A_2 = \begin{pmatrix} 1 & 5/2 \\ 5/2 & 15/2 \end{pmatrix} . \tag{6}$$

But

$$f_3(\lambda A_1 + (1 - \lambda)A_2) = 5/4 \quad , \tag{7}$$

while

$$\lambda f_3(A_1) + (1 - \lambda)f_3(A_2) = 1 \quad . \tag{8}$$

From (7) and (8) we observe that for the two positive definite matrices A_1 and A_2

$$f_3(\lambda A_1 + (1 - \lambda)A_2) > \lambda f_3(A_1) + (1 - \lambda)f_3(A_2) \tag{9}$$

for the particular λ chosen so that $0 \leq \lambda \leq 1$. Then f_3 is not a convex function.

The importance of the above discussion to the problem at hand lies in the fact that the usual convex programming procedures require the region S to be specified by a set of convex functions. In particular the algorithm of Hartley and Hocking (1963) contains this requirement.

There is, however, another way to specify the condition that the matrix B be positive semi-definite. The linear conditions on the β_{ij} given by

$$v'Bv \geq 0 \tag{10}$$

for all k-vectors v such that $v'v = 1$ also specifies that B is positive semi-definite. The description of S is in terms of simpler, linear functions of the β_{ij} but carries with it the apparent disadvantage that the number of such functions required to specify S is infinite. Hence with this description the standard quadratic programming techniques do not apply. It will be shown, however, that the Hartley-Hocking algorithm is singularly unperturbed by such an infinity of constraints, so that the problem will be formulated in section II.C with the restrictions specified by (10).

B. The Convexity of the Restraint Space S

A point in the $\binom{k+1}{2}$-dimensional space of the β_{ij}, i,j = 1,...,k may be represented by a symmetric kxk matrix $B = (\beta_{ij})$. To establish the convexity of the subset S consisting of those points for which B is positive semi-definite, it suffices to show that if B_1 and B_2 denote two points in S then $B_3 = \lambda B_1 + (1 - \lambda)B_2$ is in S for any $0 \leq \lambda \leq 1$ (see Hadley (1964)). Now B_3 is in S if and only if $v'B_3v \geq 0$ for any k-vector v. But this follows immediately since

$$v'B_3v = \lambda v'B_1v + (1 - \lambda)v'B_2v \tag{11}$$

and both $v'B_1v$ and $v'B_2v$ are non-negative.

C. Formulation in a Convex Programming Context

In order to regard (10) as a finite set of linear inequality restrictions we temporarily consider only those vectors v generated by a fine grid of space angles. Since the finiteness of this set of vectors will later be dropped we need not be more specific.

The estimation problem then requires the minimization of the quadratic $Q(\beta)$ subject to the large number of linear restrictions (10). Thus, though we lack the usual condition that all variables lie in the positive quadrant, this is just a quadratic programming problem although for any reasonable grid the number of restrictions in (10) would be extremely large. In what follows it is shown that by employing the method of 'Tangential Approximation' for convex programming (Hartley and Hocking (1963) with a special pricing operation the specification of the grid size can be completely avoided and, more importantly, only a small number of the linear restrictions $v'Bv \geq 0$ will have to be formed. Furthermore these restrictions will be formed only when needed, as specified by the algorithm.

An initial basis is required for the Simplex-like algorithm to be used. This is achieved by adjoining the restrictions $\beta_{ij} \geq -\mu$, $i,j = 0, \ldots, k$ for some large μ which must be specified. Thus the problem proposed for solution is

minimize $Q(\beta)$

subject to

$$v'Bv \geq 0 \tag{12}$$

$$\beta_{ij} + \mu \geq 0, \; i,j = 0, \ldots, k \; .$$

In the Hartley and Hocking paper an algorithm is given for solving such convex, in this case quadratic, programming problems. The algorithm proposes (i) a linearization of the original problem, (ii) reverting to the dual linear problem, and (iii) employing a special pricing operation with the revised simplex method. The essential feature of the algorithm is that the linearization of the problem need not be done in advance but only as specified by the pricing operation. For completeness, two basic points of the algorithm are reviewed here in terms of the problem (12).

The first feature of the algorithm is a linearization which is accomplished as follows. Introduce the new variable z defined by $z = -Q(\beta)$ and replace problem (12) by the equivalent problem

$$\begin{aligned} &\text{maximize } z \\ &\text{subject to} \\ &\qquad v'Bv \geq 0 \\ &\beta_{ij} + \mu \geq 0 \quad , \; i,j = 0, \ldots , k \\ &Q(\beta) + z \leq 0 \quad . \end{aligned} \tag{13}$$

The linearization of problem (13) is now completed by replacing the convex restriction $Q(\beta) + z \leq 0$ by the set of tangent planes of the form

$$Q(\beta^*) + \sum_{i \leq j=0}^{k} \partial Q(\beta^*)/\partial \beta_{ij} \; (\beta_{ij} - \beta^*_{ij}) + z \leq 0 \tag{14}$$

where the points β^* are as yet unspecified but are conceptually the points of a fine grid imposed on the $\binom{k+2}{2}$-dimensional β-space. The partial derivatives are obtained from (1) as

$$\partial Q(\beta)/\partial \beta_{ij} = -2 \sum_{t=1}^{N} (y_t - E(y_t))\,\partial E(y_t)/\partial \beta_{ij} \tag{15}$$

where

$$\partial E(y_t)/\partial \beta_{ij} = \begin{cases} 1 & i = j = 0 \\ x_{it} & j = 0 \\ 2x_{it}x_{jt} & 0 < i < j \\ x_{it}^2 & i = j > 0 \quad . \end{cases} \tag{16}$$

For computational convenience it should be pointed out that

$$Q(\beta^*)/\partial\beta_{ij} = -2 \text{ Res}_{ij} \qquad , \tag{17}$$

where Res_{ij} is the difference between the right and left sides of the $(ij)^{th}$ normal equation for the regression model I(1) when the left side is evaluated at β^*.

The second point of the algorithm which warrents a review is that of the use of the dual problem. The problem (13) with the restriction $Q(\beta) + z \leq 0$ replaced by the large set of linear restrictions (14) is now a linear programming problem having associated with it a dual linear problem, see Gass (1964), which will be solved. Rather than develop a cumbersome notation it seems better to display the dual problem in a linear programming tableau. For this purpose it is convenient to think of the $\binom{k+2}{2}$ regression coefficients β_{ij} as being numbered from 1 to $n = \binom{k+2}{2}$ in the following order

$$(\beta_{00}, \beta_{10}, \ldots, \beta_{k0}, \beta_{11}, \ldots, \beta_{1k}, \ldots, \beta_{kk}) \quad . \tag{18}$$

The tableau in Table 1 is symbolic in the sense that columns 1 and 2 simply give the rules for generating a tangent plane restriction of the form (14) for given β^* or of the form (10) for given vector v. Thus rows 1 through n+1 in columns 1 and 2 are just the coefficients of the variables β_{ij} and z in the linear restrictions (14) and (10). Row 0 of the tableau is just the negative of the constant term in the linear restrictions. Columns S_0 through S_{n+1} are self-explanatory.

	0	1	2	S_0	S_1	...	S_n	S_{n+1}
0	0	$Q(\beta^*) - \sum_{i\leq j=0}^{k} \frac{\partial Q(\beta^*)}{\partial \beta_{ij}} \beta^*_{ij}$	0	1	$-\mu$		$-\mu$	0
1	0	$\partial Q(\beta^*)/\partial \beta_{00}$	0	0	-1		0	
.	.	.	.		0			
.	.	.	.	.	.		.	.
.	.	.	.					
k+1	0	$\partial Q(\beta^*)/\partial \beta_{k0}$	0	.	.		.	.
k+2	0	$\partial Q(\beta^*)/\partial \beta_{11}$	$-v_1^2$					
.	.		$-2v_1v_2$ $\vdots$	.	.		.	.
.	.		$-2v_1v_k$					
.	.		$-v_2^2$ $\vdots$					
n	0	$\partial Q(\beta^*)/\partial \beta_{kk}$	$-v_k^2$	0	0		-1	
n+1	1	1	0	0	0		0	1

Table 1. TABLEAU FOR CONVEX PROGRAMMING

D. Solution of the Problem

Either the original linear problem or the dual problem described by the tableau of Table 1 can theoretically be solved by the Simplex method. It is clear, however, that even for small problems and reasonable grid spacings on the β-space and on the space angles to generate linear restrictions of the type (14) and (10) the number of restrictions in the original problem, or else the number of columns in the dual problem, will be extremely large.

In this section it will be shown that by solving the dual problem by the revised Simplex method with special 'pricing operations' the actual formation of the tableau is avoided. An understanding of the Simplex method is assumed and the emphasis will be on the special pricing operations. For information on the simplex method see Gass (1964) or Dantzig (1963).

At any stage of the simplex iteration, say the s^{th}, a basis matrix, say A_s consisting of n+2 columns from the tableau, is required. More precisely its inverse A_s^{-1} is required. To start the iteration the matrix A_0 consisting of columns $S_0, S_1, \ldots, S_{n+1}$ is used. It is clear that $A_0^{-1} = A_0$.

Assuming that the s^{th} stage of the iteration has been reached the matrix A_s^{-1} is available. The simplex method must now determine if any column of the tableau is eligible to 'come-into' the basis replacing one of the current columns and hence yielding a new basis A_{s+1}. The usual computation required for this step is that the scalar product of the first row of A_s^{-1}, called the pricing vector, with any column from the tableau is formed. If the result of this pricing operation is positive then the column is eligible to come-into the basis. For columns of the type S_1 through S_{n+1} this presents no problem, and column S_0 is always in the basis. However, the remaining columns are not explicitly formed and so a special pricing operation must be used.

It is shown in Hartley and Hocking (1963) that among all the vectors which could be formed by applying the rules in column 1, the one for which the pricing operation yields the largest value is just that one for which β^*_{ij}, in the order given by (18), are given by the corresponding elements of the pricing vector. That is, if the pricing vector is designated by

$$(1, p_1, \ldots, p_n, p_{n+1}) \quad (19)$$

then let $\beta^*_{00} = p_1$, $\beta^*_{10} = p_2$, ..., $\beta^*_{kk} = p_n$. Further, it is shown that the scalar product of the pricing vector with the vector from column 1 yields

$$p_{n+1} + Q(p_1, \ldots, p_n) \ . \quad (20)$$

Thus the special pricing operation to be applied to the set of columns corresponding to the tangent planes requires only the evaluation of (20) for the current pricing vector. If (20) is positive the column with β^*_{ij} given by (19) is generated and brought into the basis by the usual simplex iteration.

Now consider the problem of determining whether or not any column of the type 2 is elegible to come-into the basis. Denoting the elements p_{k+2} through p_n of the pricing vector by β^*_{ij} with the correspondence described above, the usual pricing operation applied to a column of type 2 for arbitrary v yields

$$-\sum_{ij}\sum \beta^*_{ij} v_i v_j \quad . \tag{21}$$

Denoting the symmetric matrix of β^*_{ij} by B*, it is apparent that the vector is eligible to come-in if

$$\sum_{ij}\sum \beta^*_{ij} v_i v_j = v'B^*v \tag{22}$$

is smaller than zero.

It is known (Courant and Hilbert (1953)) that the smallest value of (22) among all v such that $v'v = 1$ is given by the smallest characteristic root of the matrix B*, say λ_1. Thus if λ_1 is negative (21) is positive and at least one column of type 2 is eligible to come-into the basis. It is also known that (22) takes on the value λ_1 when v is the normalized characteristic vector corresponding to λ_1. These assertions are proved following.

To determine the vector v such that $v'v = 1$ and $v'B^*v$ is a minimum, form the Lagrangian

$$L(v,\lambda) = v'B^*v - \lambda v'v \quad . \tag{23}$$

Then set

$$\partial L(v,\lambda)/\partial v = B^*v - \lambda v = 0 \ , \tag{24}$$

or

$$(B^* - \lambda I)v = 0 \ . \tag{25}$$

Then λ is certainly one of the characteristic roots of B*, and v is the corresponding characteristic vector. But from (24)

$$v'B^*v = v'\lambda v = \lambda v'v = \lambda \ , \tag{26}$$

so that

$$\min v'B^*v = \min \lambda = \lambda_1 \quad . \tag{27}$$

Thus the special pricing operation to be applied to columns of type 2 consists of determining the smallest characteristic root, λ_1, of the matrix B^*. If λ_1 is positive then none of these columns is eligible. If λ_1 is negative then the normalized characteristic vector v corresponding to λ_1 is determined and this vector is used to generate an incoming column according to the rules in column 2 of the tableau.

When none of the pricing operations succeeds in finding a column eligible to come-into the basis the iteration is terminated. The current pricing vector then yields the desired estimates of the regression coefficients, again using the correspondence described by (19), with $-p_{n+1}$ giving the optimal value of $Q(\beta)$.

E. Computational Considerations

It will usually be worthwhile to first compute the solution to the unrestrained least squares problem. This can be done by one of the following two methods of which the first will usually be preferable:

(i) Solve the system of $\binom{k+2}{2}$ linear normal equations of the unrestrained least squares problem (see (35) below).

(ii) Solve the above linear programming problem ignoring column 2 of the tableau but terminating when $p_{n+1} + Q(p_1,\ldots,p_n) < \varepsilon$ for some moderately small positive positive ε. The pricing vector at this point yields, with accuracy depending on the choice of ε, the unrestrained least squares estimates of the β_{ij}.

If the unrestrained solution yields a positive semi-definite form no further work is needed. If not, the unrestrained solution will provide a useful preselected basis matrix in the linear programming process using the full tableau.

In case (ii) the linear programming cycles are, therefore, simply continued, inspecting both columns 2 and 1 in the pricing operation. In case (i), however, the unrestrained solution must be used to specially compute a preselected system of n+2 column vectors for the basis. The construction of this preselected basis matrix, say A_0^*, is achieved by forming a set of tangent planes to the convex surface $Q(\beta)$ in the neighborhood of the unrestrained solution, say $\hat{\beta}$. A_0^* will consist of columns S_0 and S_{n+1} from the tableau separated by n columns formed by computing column 1 at n appropriate points β^*. A detailed procedure for constructing A_0^* is discussed in the example problem at the end of this chapter.

This approach requires of course that A_0^* be non-singular. There does exist the possibility that the surface will be rather 'flat' in some large δ-neighborhood of $\hat{\beta}$ or that some of the n points β^* at which the tangent planes are constructed are selected too near $\hat{\beta}$. Both of these situations would yield tangent planes which are essentially parallel and consequently a singular A_0^*. Also the inverse of this matrix is demanded by the algorithm and must be computed.

Experience with the application of the algorithm to small trial problems indicates that procedure (i) should substantially reduce the number of linear programming cycles required for solution. However, a slight risk of encountering excessive preliminary calculations, because of a singular A_0^*, must be accepted. These calculations are again discussed in the example problem to follow.

F. Properties of the Estimates

The algorithm discussed above describes a method for finding a point estimate of the regression coefficients satisfying a convex restriction. Following is a summary of some statistical properties of such estimates.

1. The minimization of the residual sum of squares, $Q(\beta)$, in the convex region S is identical with determining that point β^* in S which is 'nearest' to the least squares estimator $\hat{\beta}$. The concept of 'nearest' refers to the metric in which the elements of $\hat{\beta}$ are independently distributed with equal variance. (Lewish (1963)). This result will be derived as a starting point for the discussion in Chapter IV.

2. If S is convex and the true parameter β is in S, then

$$(\beta^* - \beta)'(\beta^* - \beta) \leq (\hat{\beta} - \beta)'(\hat{\beta} - \beta) \quad \text{(Lewish (1963)} \quad (28)$$

3. As a consequence of property 2,

$$E(\beta^* - \beta)'(\beta^* - \beta) \leq E(\hat{\beta} - \beta)'(\hat{\beta} - \beta) \quad , \qquad (29)$$

or

$$\sum_{i=1}^{n} MSE(\beta_i^*) \leq \sum_{i=1}^{n} Var(\hat{\beta}_i) \qquad \text{(Lewish (1963))} \quad . \quad (30)$$

4. The point estimates are clearly maximum likelihood estimates since the likelihood is proportional to $\exp\{-Q(\beta)/2\sigma^2\}$ where $Q(\beta)$ is minimized within the restricted parameter space $\beta \epsilon S$. The estimates are

then consistent since this is a property associated with restricted maximum likelihood estimation; see Kendall and Stuart (1961).

5. The estimators are functions of a minimal set of sufficient statistics. To show this write $Q(\beta)$ in the form

$$Q(\beta) = \text{Reg}(\hat{\beta} - \beta) + \text{Res}(Y) \tag{31}$$

where $\text{Reg}(\hat{\beta} - \beta) = (\hat{\beta} - \beta)'X'X(\hat{\beta} - \beta)$ is the classical regression component involving only the unrestricted least squares estimator $\hat{\beta}$ and $\text{Res}\ (Y) = (Y - X\hat{\beta})'(Y - X\hat{\beta})$ is the 'residual' which does not involve β. The minimum of $Q(\beta)$ in S is, therefore, attained at a parameter point which only depends on $\hat{\beta}$ and the latter represent a minimal set of sufficient statistics. There then result the optimality properties based on minimal sufficiency; see Rao (1965).

6. An exact confidence region with confidence coefficient $1 - \alpha$ can be computed as follows: Consider the customary confidence region R given by

$$\text{Reg}(\hat{\beta} - \beta) \leq n\text{Res}(Y)^{F(\alpha;n,N-n)/(N-n)} \quad . \tag{32}$$

In the present case an exact confidence region for β is then clearly given by the intersection of S and R, i.e., by $\beta \epsilon S\ R$. (In case the intersection is empty no statement about β will be made.) Since this confidence region is based on minimal sufficient statistics it enjoys the properties described by the 'intersection principle' introduced by Roy and Bose (1953). (This region forms the basis for the results in Chapter III.)

7. Further properties are described by Hartley (1963) and the distribution in the case of a single relevant restriction has been

derived by Hocking (1965).

It should be mentioned that the above properties were not developed for the particular estimator β^* derived in this chapter. Rather they are properties depending only on the convexity of a restraint space and are enjoyed by any such estimator.

G. A Numerical Example

As an example to illustrate the algorithm the following problem is considered. It is desired to estimate the coefficients in the model

$$E(y_t) = \beta_{00} + \beta_{10}x_{1t} + \beta_{20}x_{2t} + \beta_{11}x_{1t}^2 + 2\beta_{12}x_{1t}x_{2t} + \beta_{22}x_{2t}^2 \tag{33}$$

subject to the restriction that the matrix

$$B = \begin{pmatrix} \beta_{11} & \beta_{12} \\ \beta_{12} & \beta_{22} \end{pmatrix} \tag{34}$$

be positive semi-definite.

A central composite design yielding 9 data points was selected for estimating the 6 coefficients of this second-order response surface. The date are shown in Table 2.

t	x_{1t}	x_{2t}	y_t
1	-2	0	0.8
2	-1	-1	13.9
3	-1	1	10.1
4	0	-2	41.8
5	0	0	2.0
6	0	2	42.2
7	1	-1	9.7
8	1	1	13.9
9	2	0	1.4

Table 2. Data

The unrestricted least squares solution, ignoring the restriction (34), was obtained by solving the normal equations

$$X'X\hat{\beta} = X'Y \tag{35}$$

where

$$X'X = \begin{bmatrix} 9 & 0 & 0 & 12 & 0 & 12 \\ 0 & 12 & 0 & 0 & 0 & 0 \\ 0 & 0 & 12 & 0 & 0 & 0 \\ 12 & 0 & 0 & 36 & 0 & 4 \\ 0 & 0 & 0 & 0 & 16 & 0 \\ 12 & 0 & 0 & 4 & 0 & 36 \end{bmatrix} , \tag{36}$$

$$(X'Y)' = (135.8,0.8,1.2,56.4,16.0,383.6) , \tag{37}$$

and

$$\hat{\beta}' = (b_{00},b_{10},b_{20},b_{11},b_{12},b_{22}) . \tag{38}$$

The solution is

$$\hat{\beta}' = (2.112,\ 0.067,\ 0.100,\ -2.246,\ 1.000,\ 9.979) .\tag{39}$$

The negative estimate of β_{11} violates restriction (34) and requires that the convex programming algorithm be employed. Then let

$$z = -Q(\beta) = -\sum_{t=1}^{9} \{y_t - E(y_t)\}^2 \tag{40}$$

and let

$$\mu = 10. \tag{41}$$

The convex programming problem is

maximize z

subject to $\quad v'Bv \geq 0$

$$\beta_{ij} + 10 \geq 0 \text{ , } i,j = 0,1,2 \tag{43}$$

$$z + Q(\beta) \leq 0 \quad .$$

The tableau for this example is spelled out in detail in Table 3. The vector β^* used to generate a typical column 1 if desired is obtained from the pricing vector for the current iteration with the correspondence described in (19). The vector (v_1,v_2) used to generate a typical column 2 if desired is just the normalized characteristic vector corresponding to the minimum characteristic root of the current B matrix given by

$$B^* = \begin{pmatrix} \beta^*_{11} & \beta^*_{12} \\ \beta^*_{12} & \beta^*_{22} \end{pmatrix} \quad . \tag{43}$$

A comment on the choice of the constant μ is also in order. The constant must be chosen so that the optimum value of every β_{ij} exceeds $-\mu$. Thus a large positive μ is suggested. But if μ is chosen to be extremely large in comparison with the size expected for the coefficients the number of linear programming cycles is greatly increased. Here it was decided from observation of the least squares solution (39) that $\mu = 10$ should be satisfactory.

The positive semi-definite restriction on B includes the restrictions $\beta_{11} \geq 0$ and $\beta_{22} \geq 0$ so that the restrictions $\beta_{11} \geq -\mu$, $\beta_{22} \geq -\mu$ are not necessary. This is reflected in columns S_4 and S_6. Similarly the first element of column S_7, in general S_{n+1}, may be replaced by any legitimate lower bound on $Q(\beta)$. In this case, since the unrestrained minimum of $Q(\beta)$, namely $Q(\hat{\beta}) = 0.194$, is available from the least squares solution, it was used.

Col / Row	0	1	2	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
0	0	$\sum_{t=1}^{9} y_t^2 - \beta^{*\prime}(X'X)\beta^*$	0	1	-10	-10	-10	0	-10	0	.194
1	0	$2\{9\beta^*_{00} + 12\beta^*_{11} + 12\beta^*_{22} - 135.8\}$	0	0	-1	0	0	0	0	0	0
2	0	$2\{12\beta^*_{10} - .8\}$	0	0	0	-1	0	0	0	0	0
3	0	$2\{12\beta^*_{20} - 1.2\}$	0	0	0	0	-1	0	0	0	0
4	0	$2\{12\beta^*_{00} + 36\beta^*_{11} + 4\beta^*_{22} - 56.4\}$	$-v_1^2$	0	0	0	0	-1	0	0	0
5	0	$2\{16\beta^*_{12} - 16\}$	$-2v_1v_2$	0	0	0	0	0	-1	0	0
6	0	$2\{12\beta^*_{00} + 4\beta^*_{11} + 36\beta^*_{22} - 383.6\}$	$-v_2^2$	0	0	0	0	0	0	-1	0
7	1	1	0	0	0	0	0	0	0	0	1

Table 3. Example Tableau

The vectors $S_1, \ldots, S_6$, in general $S_1, \ldots, S_n$, correspond to the restrictions $\beta_{ij} \geq -\mu$, $i \neq j$, and $\beta_{11} \geq 0$, $\beta_{22} \geq 0$, and are used only to obtain a starting basis if none is available. They are only shown for completeness here as they were not used in this example. Instead, the unrestrained optimum (39) was used to generate an initial basis using $n = \binom{k+2}{2} = 6$ columns of type 1 computed using β^*_{ij} the values

$$\beta^*_{i'j'} = b_{i'j'} + \varepsilon_2/\sqrt{c_{tt}} \tag{44}$$

for a particular pair of subscripts $i'j'$ with

$$\beta^*_{ij} = b_{ij} \tag{45}$$

for all remaining coefficients, where c_{tt} is the t^{th} diagonal element of $X'X$, $\beta^*_{i'j'}$ represents the element in the t^{th} position of the vector β^*, and ε_2 was empirically chosen equal to 1. The six tangential plane columns arising from this substitution for $t = 1,\ldots,6$ were then monitered in the computer for rank-degeneracy. In case rank-degeneracy had been found the program (see Chapter III) provides for increasing the value of ε_2 sequentially by a unit at a time, but for this example an acceptable basis was found with $\varepsilon_2 = 1$. Although this procedure commences from the empirical formula ((44),(45)) it will always provide an acceptable preselection and any shortcomings of this formula will merely increase the number of cycles in the linear programming process. It is expected, however, that such increase would be slight. Excessive computations resulting from rank-degeneracy would appear to be the larger drawback of the procedure, although such a situation would arise only rarely. In the event of such an undesirable situation one may of course resort to the method (ii) of section E, beginning the linear programming iterations with the basis consisting of columns S_0 through S_7 of the tableau.

The 6 columns generated from formula ((44),(45)) along with columns S_0 and S_7 of the tableau constituted A_0^*. The inverse of this matrix was then computed and the linear programming iterations were begun. Table 4 shows the solution of the example, exhibiting the initial β^* from A_0^{*-1}, the result of every 5^{th} iteration, and the solution. Termination of the iteration occured when neither $p_7 + Q(p_1,..$ $Q(p_1,...,p_6)$ nor λ_1 exceeded $\varepsilon_1 > 0$, where ε_1 was chosen to be .001.

The solution as presented in table 4 prompts the following comments. The choice of ε_2 gave column 2 a chance to come-in early here coming into the basis on iteration number 15. Then between iterations 20 and 25 a tangent plane of the column 1 type came in to replace column S_7 with rather dramatic effect. Beyond this stage it is seen that the algorithm works quite diligently to bring $-p_7$ and Q together, all the while keeping the matrix B near definiteness. It is also clear that the number of cycles required to solution is very sensitive to the choice of ε_1. Finally, the estimates of β_{11} and β_{22} are observed to be positive, and the determinant of B is -.0000029, or effectively zero.

This example affords a comparison between the procedures (i) and (ii) of section E. Using (ii), which simply solves the problem by starting with the basis $S_0,S_1,...,S_7$, a total of 155 iterations were required. Using (i) the unrestrained solutions were used to construct an initial feasible basis as described in the example and only 80

TABLE 4

Solution of Example

Iteration Number	P_1	P_2	P_3	P_4	P_5	P_6	$-P_7$	Q
(From $A^{*-1}_{0\ 0}$)	1.347773	.233553	.266553	.081414	1.499972	10.306414	---	----
5	1.738254	.233456	.266456	-.064625	1.190407	10.159443	.194000	2.070711
10	2.583632	.233245	.144876	-.389432	1.015632	9.849895	.194000	.962953
15	1.207977	.039473	.108480	.142834	1.248287	10.350190	.194000	3.612505
20	.882988	-.132143	-.047069	.105043	1.040852	10.244894	.194000	4.075216
25	3.440911	-.422496	-1.439664	-.033088	.240982	8.966571	.558760	67.936867
30	1.026051	.075972	.235134	.099474	1.013246	10.292798	.655206	2.553969
35	1.638020	-.003707	-.004751	.083045	.916986	10.116644	1.219540	2.150959
40	1.381339	.027651	.165868	.089611	.955306	10.183857	1.444102	1.738089
45	1.393151	.110643	.076891	.095199	.984033	10.157754	1.525267	1.736052
50	1.463203	.007878	.067221	.084341	.924254	10.123599	1.592946	1.795668
55	1.508592	.074923	.127114	.086674	.936929	10.127281	1.617844	1.748942
60	1.372265	.079259	.083392	.095815	.988720	10.187396	1.645202	1.696255
65	1.405463	.079300	.100507	.090187	.958248	10.180682	1.657626	1.670604
70	1.393125	.068104	.090332	.089927	.956650	10.176368	1.663322	1.669515
75	1.390199	.063924	.104766	.088501	.949497	10.186827	1.665030	1.667416
80	1.405676	.065765	.108597	.088602	.949611	10.177615	1.665903	1.667940
(Solution) 81	1.400070	.067797	.102001	.088921	.951372	10.178818	1.666003	1.666761

iterations were required. Further acceleration might be made by improving upon the method of construction of the initial basis, but it is doubtful that a substantial gain would be made.

Another method for accelerating convergence for a problem of this size is that of expressing the coefficients of the linear terms in (33) as linear functions of the elements of the B-matrix (34). This is easily done by considering the unrestricted minimization of the quadratic form

$$\{(Y - X'BX) - b'X\}'\{(Y - Y'BX) - b'X\} \tag{46}$$

considered as a function of the elements of b for a given B. The solution is

$$\hat{b} = (X'X)^{-1}X'(Y - BX). \tag{47}$$

The result of this initial calculation is the reduction of the number of coefficients which must be estimated from 6 to only 3. Consequently the size of the tableau is reduced roughly by half and the number of iterations required for solution might be decreased accordingly. This approach, however, has not been pursued since its advantage is reduced as the number of coefficients in the problem is increased, and even that advantage may disappear when the amount of requisite preliminary calculation is considered.

CHAPTER III

THE COMPUTER PROGRAM FOR MODEL SAMPLING

The computer program for the restricted estimation problem in Chapter II was initially written for the IBM 7094 computer on the campus of Texas A&M University and was an extension of a program by Claypool (1966). Since the problem of model sampling necessarily includes the estimation problem, only the program for model sampling is herein exhibited. The subroutines INVEC, NETPRC, OUTVEC, and BNVERS were abstracted in their entirety from Claypool's original program. The remainder of the program, however, is entirely the work of this author and the whole model sampling program is considered by him to be a vital part of the dissertation.

The program listed following is designed for the IBM 360, MOD 40 computer on the campus of West Texas State University, where the latter stages of research on this problem were carried out. Comments to the right of the program proper are intended to illustrate the role of the indicated piece of the program in the solution of the problem.

```
/FTC      LIST,NOMAP
C  MONTE CARLO GENERATION - DISTRIBUTION OF LAMBDA X  AND U X
C
C
      COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
     1  NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
     2  RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
     3  EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
     4  CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
     5  IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
      READ 5,2  M,N,NUM,KEY,AR,EPS1,EPS2,EPS3
    2 FORMAT  4I5,4F10.4
      KAGAIN   1
    3 ITR   0
      IPAGE   1
      NLI   0
      ITER   0
      ITERA   0
      IHALT   0
      ISTOP   0
      ISWT   0
      IHOPE 0
      MP1   M 1
      MP2   M 2
      NP1   N 1
      NP2   N 2
      MN1   M N 1
      ROOT   1.
      NEIGEN   0
      DRHOH   0.
      IHOCK   0
    5 CALL CDATA
      IF DRHOH  5,5,6        Goes to cycle n if cycle n-1 yielded u(X)=0.
    6 CALL BUILDA            Begins convex programming solution for β* if
      OVER   0.              required.
```

```
      DO 4 I   1,NP2
    4 NB I    O
      DO 301 I   1,6
      CSYM I    0.
  301 CONTINUE
      DO 303 I   1,6
      DO 305 J   1,9
  305 CSYM I    CSYM I    C I,J **2
  303 CONTINUE
      DO 314 I   1,N
  314 XINITL I    BETA I
      DO 103 I   1,NP2
      X I    0.
  103 CONTINUE
  113 OVER   OVER   1.
      IF OVER-101.  114,201,201
  114 DO 105 I   1,N
      X I    XINITL I     OVER*EPS3 /SQRT CSYM I
      DO 107 K   1,N
      IF K-I  108,107,108
  108 X K    XINITL K
  107 CONTINUE
      CALL NETPRC
      CALL AMATRX
      DO 115 J   1,7
      K1   I 1
      ARRAY J,K1    A J,7
  115 ARRAY 8,K1    1.
  105 CONTINUE
      ARRAY 1,1    1.
      ARRAY 8,8    1.
      ARRAY 1,8    -BIGM
      ARRAY 8,1    0.
      DO 111 J1   2,7
      ARRAY J1,1    0.
```

Constructs A_0^* by the method on page 27.

```
      ARRAY J1,8     0.
  111 CONTINUE
      DET    1.
      DO 77 I    1,NP2
      PIV    ARRAY I,I
      DET    DET*PIV
      IF ABS DET -1.E-6  71,113,71
   71 DO 72 J    1,NP2
      IF I-J  72,73,72
   73 ARRAY I,J     1.
   72 ARRAY I,J     ARRAY I,J /PIV
      DO 77 J    1,NP2                                   Computes A*0^-1
      IF I-J  74,77,74
   74 POT    ARRAY J,I
      DO 75 K    1,NP2
      IF K-I  75,76,75
   76 ARRAY J,K     0.
   75 ARRAY J,K     ARRAY J,K -ARRAY I,K *POT
   77 CONTINUE
      DO 120 I    1,NP2
      DO 110 J    1,NP2
  110 B I,J    ARRAY I,J
  120 CONTINUE
      DO 130 I    1,NP1
      I3    I    1                                       Gives current
      X I     B 1,I3                                       solution
  130 CONTINUE
      GO TO 203
  201 WRITE  6,205
  205 FORMAT 10X, ONE HUNDRED TRIALS YIELDED NO B-INVERSE.  Stops
      STOP                                                 program if A*0 remains
  203 IHOPE    1                                           singular at 100EPS3.
   14 ITER    ITER    1
      ITERA    ITERA    1
      CALL INVEC
```

```
      IF IHALT  42,15,42            Performs the linear programming
   15 IF  ISTOP  40,16,40              iterations.
   16 CALL OUTVEC
      IF  ISTOP  40,18,40
   18 CALL BNVERS
      DO 501 I   1,9
      DD I     YY I
      DO 503 J  1,6
  503 DD I     DD I   - C I,J *X J
  501 CONTINUE
      QB    0.
      DO 505 I   1,9
      QB    QB    DD I **2
  505 CONTINUE
      GO TO 14
   42 IF  KEY  40,39,40
   39 XLAM    EXP -.5* QB-QBU
      XLOGX   -2.*ALOG XLAM
      WRITE 6,300  XLAM,XLOGX        Computes and writes λ(X) and u(X).
  300 FORMAT 40X,2E17.8/                (if u(X) ≠ 0)
      IHOPE 0
      GO TO 5
   40 STOP
      END
/FTC     LIST,NOMAP
      SUBROUTINE INVEC
      COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
     1 NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
     2 RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
     3 EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
     4 CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
     5 IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
C WE ARE LOOKING FOR K SUCH THAT MAX C J  - Z J    C K  - Z K  IS
C GREATER THAN ZERO
      SMAX   0.
```

```
      SMAX1   0.
      CALL EIGNET
      CALL NETPRC
      DO 12 I1   1,MN1
      IF  ZNET I1  - EPS2  2,2,4
    2 IF  ZNET I1   12,12,8
    4 IF  SMAX - ZNET I1   6,12,12
    6 SMAX   ZNET I1
      K   I1
      GO TO 12
    8 IF  SMAX1 - ZNET I1   10,12,12
   10 SMAX1   ZNET I1
      K1   I1
   12 CONTINUE
      IF  SMAX  18,14,26
   14 IF  SMAX1  18,22,16
   16 IHALT   1
      GO TO 30
   18 WRITE  6,20
   20 FORMAT   1 ,10X, IMPOSSIBLE STOP AT SUBROUTINE INVEC.
      ISTOP   1
      GO TO 30
   22 IHALT   1
      GO TO 30
C  THE E I  CONSTITUTE THE COLUMN VECTOR TO ENTER BASIS,  THIS REPRESEN-
C  TS COLUMN -INV- OF THE -A-MATRIX.
   26 INV   K
      CALL AMATRX
      DO 28 I   1,NP2
   28 E I    A I,K
      IF   INV - MN1  30,29,30
   29 NLI   NLI   1
   30 RETURN
      END
/FTC     LIST,NOMAP
```

```
      SUBROUTINE NETPRC
      COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
     1 NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
     2 RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
     3 EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
     4 CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
     5 IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
C  THIS SUBROUTINE CALCULATES THE SMALLEST NET PRICE FOR THE I-TH SET
C  OF RESTRICTIONS AND MUST BE SUBMITTED INDEPENDENTLY FOR EACH PROBLEM
      ZNET 1    -X 1  - 10.
      ZNET  2    -X 2  - 10.
      ZNET 3    -X 3  - 10.
      ZNET 4    -X 4
      ZNET 5    -X 5  - 10.
      ZNET 6    -X 6
      DO 8 I   1,9
      DD I     YY I
      DO 4 J   1,6
    4 DD I     DD I   - C I,J *X J
    8 CONTINUE
      QB   0.
      DO 10 I   1,9
   10 QB   QB   DD I **2
      ZNET 7    X 7    QB
      ZNET 8    -ROOT
   40 RETURN
      END
/FTC     LIST,NOMAP
      SUBROUTINE AMATRX
      COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
     1 NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
     2 RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
     3 EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
     4 CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
     5 IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
```

```
C   THIS SUBROUTINE MUST BE SUBMITTED INDEPENDENTLY FOR EACH PROBLEM.
       IF IHOPE  2,2,17
   17 IF  NP1 - INV  40,2,40
    2 DO 4 I   2,7
       A I,7   0.
       K   I - 1
       DO 6 J   1,9
    6 A I,7    A I,7    C J,K *DD J
    4 A I,7    -2.*A I,7
       A 1,7   QB
       DO 8 I   2,7
       K   I - 1
    8 A 1,7    A 1,7  - A I,7 *X K
   40 RETURN
       END
/FTC      LIST,NOMAP
       SUBROUTINE OUTVEC
       COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
     1  NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
     2  RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
     3  EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
     4  CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
     5  IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
C   THE UPDATED P 0  VECTOR  BINV*P 0   IS LAST COLUMN OF B INVERSE,
C   ELEMENTS B I,NP2 .  THE UPDATED -E- VECTOR  BINV*E  HAS ELEMENTS
C   D I .
       DO 4 I   1,NP2
       D I    0.
       DO 2 J   1,NP2
    2 D I    D I    B I,J *E J
    4 CONTINUE
C   FOR D I  GREATER THAN ZERO, FIND THE MINIMUM NON-NEGATIVE RATIO
C   B I,NP2 /D I .
       K   0
       K1   0
```

1. Constructs 6 columns of A_0^* prior to linear programming iterations.
2. During linear programming iterations this obtains column 1 as indicated on page 27.

```
      SM1    1.
      DMAX    0.
      RMIN    10.**10
      DO 14 I    2,NP2
      IF  D I    6,14,6
    6 RAT    B I,NP2 /D I
      IF  RAT  14,8,10
    8 IF  D I  - DMAX  14,14,9
    9 DMAX    D I
      RMIN    0.
      K1    I
      GO TO 14
   10 IF  RMIN - RAT  14,14,12
   12 RMIN    RAT
      K1    I
   14 CONTINUE
      IF  K1  16,16,20
   16 WRITE  6,18
   18 FORMAT   1 ,10X, THERE IS NO BOUNDED SOLUTION
      ISTOP   1
      GO TO 24
C  COLUMN -IOUT- IS REPLACED IN BASIS.
   20 IOUT   K1
      NB IOUT     INV
      DO 22 I    1,NP2
   22 BAS I,IOUT     E I
   24 RETURN
      END
/FTC     LIST,NOMAP
      SUBROUTINE BNVERS
      COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
     1 NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
     2 RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
     3 EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
     4 CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
```

```
     5  IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
C  UPDATE B-INVERSE MATRIX BY PREMULTIPLYING BY THE ELEMENTARY MATRIX
C   EL I,J  .
      DO 4 I   1,NP2
      DO 2 J   1,NP2
    2 EL I,J    0.
    4 CONTINUE
      DO 6 I   1,NP2
    6 EL I,I    1.
      DO 12 I   1,NP2
      IF  I - IOUT  10,8,10
    8 EL I,I    1./D IOUT
      GO TO 12
   10 EL I,IOUT    -D I /D IOUT
   12 CONTINUE
      DO 18 I   1,NP2
      DO 16 J   1,NP2
      BA I,J    0.
      DO 14 K   1,NP2
   14 BA I,J    BA I,J    EL I,K *B K,J
   16 CONTINUE
   18 CONTINUE
      DO 22 I   1,NP2
      DO 20 J   1,NP2
   20 B I,J   BA I,J
   22 CONTINUE
C  FROM B-INVERSE WE OBTAIN THE CURRENT SOLUTION, X I    B 1,IP1 .
      DO 24 I   1,NP1
      IP1   I   1
   24 X I    B 1,IP1
      RETURN
      END
/FTC     LIST,NOMAP
      SUBROUTINE BUILDA
      COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
```

```
      1  NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
      2  RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
      3  EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
      4  CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
      5  IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
       DO 4 I    1,8
       DO 2 J    1,7
     2 A I,J     0.
     4 CONTINUE
       DO 6 I    1,6
       K    I    1
       A 1,I     -10.          Constructs first 7 columns of A0(page 17).
     6 A K,I     -1.
       A 8,7     1.
       A 1,4     0.
       A 1,6     0.
       RETURN
       END
/FTC      LIST,NOMAP
       SUBROUTINE CDATA
       COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
      1  NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
      2  RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
      3  EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
      4  CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
      5  IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
       IF IHOCK  56,55,56  Test to see if (X'X)-1 is known.
    55 READ 5,2     C I,J ,J 1,6 ,I 1,9 The matrix X is read in.
     2 FORMAT 12F6.2
       READ 5,4    CX I ,I 1,9  The vector (Y - e) is read in.
     4 FORMAT 9F6.2
       READ 5,6  IX    An initial random odd integer with 9 or fewer
     6 FORMAT 1I9        digits is read in.
       DO 80 I 1,6
       DO 80 K 1,6
```

```
   DO 80 J 1,9
80 XPX I,K     0.
   DO 85 I 1,6
   DO 85 K 1,6
   DO 85 J 1,9
85 XPX I,K     XPX I,K    C J,I *C J,K
   DO 77 I 1,N
   PAV   XPX I,I
   DO 72 J 1,N
   IF I-J  72,73,72                Computes and inverts X'X.
73 XPX I,J     1.
72 XPX I,J     XPX I,J /PAV
   DO 77 J 1,N
   IF I-J  74,77,74
74 PAT   XPX J,I
   DO 75 K 1,N
   IF K-I  75,76,75
76 XPX J,K     0.
75 XPX J,K     XPX J,K -XPX I,K *PAT
77 CONTINUE
56 IHOCK IHOCK 1
   IF IHOCK-101  12,13,13          Stops the program after 500 cycles.
13 STOP
12 DO 20 K 1,9
   RE K  0.
   DO 22 J 1,12
   IX IX*65539                     Generates 9 random N(0,1) variables
   IF IX  31,32,32                   for each cycle.  (Abstracted with
31 IX IX 2147483647 1                modifications from page 47, IBM
32 Y J  IX                           Programmer's Manual, System/360
   Y J  Y J *.4656613E-9             Scientific Subroutine Package).
   RE K  RE K  Y J
22 CONTINUE
21 RE K  RE K -6.0
20 CONTINUE
```

```
      DO 30 LL 1,9
   30 YY LL     CX LL     RE LL          Computes (Y - e) + e = Y
      DO 40 I 1,6
   40 XPY I     0.
      DO 50 I  1,6
      DO 50 J  1,9
   50 XPY I     XPY I     C J,I *YY J          Computes the unconstrained
      DO 90 I 1,6
   90 BETA I    0.                                 solution β̂.
      DO 100 I 1,6
      DO 100 J 1,6
  100 BETA I     BETA I   XPX I,J *XPY J
      IF BETA 4   150,160,160                            Tests for definiteness.
  160 IF BETA 6   150,170,170
  170 IF BETA 4 *BETA 6 -BETA 5 **2  150,180,180
  150 DRHOH  1.           Indicates that convex programming is required.
      DO 501 I 1,9
      DV I     YY I
      DO 503 J 1,6
  503 DV I     DV I -C I,J *BETA J       Computes Q(β̂).
  501 CONTINUE
      QBU  0.
      DO 505 I 1,9
  505 QBU   QBU DV I **2
      BIGM -QBU
      IHALT 0
      RETURN
  180 DRHOH  0.          Indicates that β̂εS.
      XLAM   1.
      XLOGX  0.
      WRITE 6,300  XLAM,XLOGX
  300 FORMAT 40X,2E17.8/            Writes λ(X) and u(X).
      IHALT 0
      RETURN
      END
```

```
/FTC      LIST,NOMAP
       SUBROUTINE EIGNET
       COMMON EPS1,EPS2,IHALT,INV,IOUT,ISTOP,ITER,ITR,M,MN1,MP1,MP2,N,
      1  NP1,NP2,SMIN,KAGAIN,N2,NLI,QB,ROOT,NEIGEN,DRHOH,IHOCK,RAO,
      2  RUDY,IHOPE,A 10,18 ,B 10,10 ,BA 10,10 ,BAS 10,10 ,D 10 ,E 10 ,
      3  EL 10,10 ,H 10 ,INT 10 ,NB 10 ,X 10 ,ZNET 18 ,C 10,10 ,Y 12 ,
      4  CX 10 ,DD 10 ,EE 10 ,ARRAY 8,8 ,XINITL 6 ,CSYM 6 ,ANT 8,16 ,
      5  IPAGE,XPX 6,6 ,RE 9 ,YY 9 ,XPY 6 ,BETA 6 ,DV 9 ,QBU,BIGM,AR
       ROOT   0.
       ROOT   ROOT     X 4    X 6   - SQRT  X 4    X 6  **2   4.*
      1 X 5 **2 - X 4 *X 6    *.5
       R1   1.
       R2    ROOT - X 4  /X 5
       VNORM   SQRT R1**2   R2**2       Performs the Column 2 operation
       V1   R1/VNORM                      indicated on page 27.
       V2   R2/VNORM
       DO 8 J   1,NP2
     8 A J,8    0.
       A 5,8    -V1**2
       A 6,8    -2.*V1*V2
       A 7,8    -V2**2
       RETURN
       END
/DATA
    1    6    0    0    .1848    .001    .001    1.
  1.  -2.   0.   4.   0.   0.   1.  -1.  -1.   1.   2.   1.
  1.  -1.   1.   1.  -2.   1.   1.   0.  -2.   0.   0.   4.
  1.   0.   0.   0.   0.   0.   1.   0.   2.   0.   0.   4.
  1.   1.  -1.   1.  -2.   1.   1.   1.   1.   1.   2.   1.
  1.   2.   0.   4.   0.   0.
  7.  11.  -3.  11.   1.   7.   1.  11.  11.
56824421
```

LIST OF REFERENCES

J. Aitchison and S. D. Silvey, "Maximum Likelihood Estimation of Parameters Subject to Restraints", Annals of Mathematical Statistics, (1958), 29: - .

E. M. L. Beale, "On Minimizing a Convex Function Subject to Linear Inequalities", Journal of the Royal Statistical Society, (1955), Series B, 17: - .

G. E. P. Box and K. B. Wilson, "On the Experimental Attainment of Optimum Conditions", Journal of the Royal Statistical Society, (1951), Series B, 13: pp. 1-45.

P. L. Claypool, Convex Programming Algorithm of Hartley and Hocking, Department of Statistics, Texas A&M University (College Station: By the author, 1966).

R. Courant and D. Hilbert, "Methods of Mathematical Physics", Interscience Publishers, (1953).

G. B. Dantzig, Programming in a Linear Structure, Comptroller, USAF (Washington, D. C., 1948).

G. B. Dantzig, Linear Programming and Extensions, (Princeton: University Press, 1963).

O. L. Davies, The Design and Analysis of Industrial Experiments, (Edinburgh: Oliver and Boyd, 1956).

S. L. Gass, Linear Programming, (New York: McGraw-Hill Book Co., Inc., 1964).

R. L. Graves and P. Wolfe (eds.), Recent Advances in Mathematical Programming, (New York: McGraw-Hill Co., Inc., 1963).

F. A. Graybill, An Introduction to Linear Statistical Models, (New York: McGraw-Hill Book Co., Inc., 1961).

G. Hadley, Nonlinear and Dynamic Programming, (Reading, Massachusetts: Addison Wesley Publishing Co., Inc., 1964).

H. O. Hartley, Applications of Digital Computers, (New York: Ginn and Company, 1963.

H. O. Hartley and R. R. Hocking, "Convex Programming by Tangential Approximation", Management Science, (1963), 9:600-612.

E. O. Heady and J. L. Dillon, Agricultural Production Functions, (Ames, Iowa: Iowa State University Press, 1961).

R. R. Hocking, "The Distribution of a Projected Least Squares Estimator", Annals of the Institute of Statistical Mathematics, (1965), 17: 357-362.

H. Hotelling, "Experimental Determination of the Maximum of a Function", Annals of Mathematical Statistics, (1941), 12: 20-45.

G. G. Judge and T. Takayama, "Inequality Restrictions in Regression Analysis", Journal of the American Statistical Association, (1966) 61: 166-181.

M. G. Kendall and A. Stuart, The Advanced Theory of Statistics, (London: Charles Griffin and Co., Ltd, Vol. II, 1961).

W. T. Lewish, Linear Estimation in Convex Parameter Spaces, Iowa State University of Science and Technology, (Ames, Iowa: By the author, 1963).

H. A. Meyer (ed.), Symposium on Monte Carlo Methods, (New York: John Wiley and Sons, Inc., 1956).

C. R. Rao, Linear Statistical Inference and Its Applications, (New York: John Wiley and Sons, Inc., 1965).

S. O. Rice, "The Distribution of the Maxima of a Random Curve", American Journal of Mathematics, (1939), 61: 409-416.

S. N. Roy and R. C. Bose, "Simultaneous Confidence Interval Estimation", Annals of Mathematical Statistics, (1953), 24: 513-538.

H. Theil, "On the Use of Incomplete Prior Information in Regression Analysis", Journal of the American Statistical Association, (1963), 58: 401-414.

T. E. Tramel, "Response Surfaces: Experimental Results with Cotton", Journal of the American Statistical Association, (1963), 58: 563.

P. Wolfe, "The Simplex Method for Quadratic Programming", Econometrica, (1959), 27: 382-398.

CONSTRAINED RUN-OUT COST ESTIMATION

With the completion of the Gemini program and the availabiliby of spacecraft systems cost data the problem of run-out cost estimation for advanced spacecraft programs is becoming an increasingly important area. The development of analytical techniques for analysis and processing of cost data are becoming increasingly complex in order to obtain a level comparable with the data which is presently available. The material presented in this paper is an extension of some earlier work which was accomplished under the NASA Research Grant. Initially the problem of estimating spacecraft run-out cost was one of using a least squares fit to a preselected set of percent cost - percent time curves which are defined on the unit interval passing through the origin and through point (1, 1). By taking generalized curve forms of this type it is then possible to take a partially completed cost - time history of a spacecraft subsystem, sort through the group of curves until the best weighted least squares fit was achieved between the partial curve and the complete curve. After this is accomplished, it is possible to make a projection of the partial curve to its completion date. This type of technique will work very well for a number of cases of cost - time histories; however, it is possible to obtain data points such that when they are applied to a standard group of third order polynomials that the run-out cost will be less than some previous cost during the course of the program. That is to say the under normal least squares methods of fitting any general set of data points to a general class of third order or high order polynomials it would be possible to obtain maxima in the range zero to one. By constraining the polynomials such that there are not maxima inside the interval zero to one it would be possible to provide run-out cost estimates which are more realistic from a real-world standpoint. It should be pointed out that a constrained least squares estimate must necessarily have a larger error sum of squares than one which is not constrained. Therefore, it is to the advantage of the analyst and to the model builder to use the minimum number and minimum level of constraints which are necessary to assure the type of general performance required from an algorithm. The following approach was developed to conform to the above requirements.

In finding an equation that best fits the data points for percent cost vs. percent completion time, some arrangement of data points could cause the slope of the equation to be negative in the region of interest, which is the decimal percent time between zero and one. This would mean that the model would be predicting that cumulative cost would decrease with time, which is hardly feasible.

Therefore the constraint that the slope of the curve must be non-negative for the domain of the function must be imposed on the model.

To apply the constraint in a continuous form for the domain of the function would needlessly burden the solution and increase the complexity of the problem. Therefore, a check of the slope for sufficiently small intervals from zero to one will suffice. The computer program to be supplied NASA - MSC checks at intervals of 0.05. To change the interval size would be a routine matter.

The check to insure a non-negative slope would require that:

$$B + 2C(X_j) + 3D(X_j)^2 \geq 0$$

$$\forall X_j \varepsilon \quad 0 \leq X_j \leq 1$$

For each point (denoted by X_j) that does not meet the inequality, a constraint term must be combined with the least square function. The term will be:

$$\lambda_j \, (B + 2CX_j + 3DX_j^2 - U_j)$$

The term U_j is a slack variable, and either λ_j or U_j must be zero, depending on the problem. Therefore each combination of U_j or λ_j equaling zero must be tried. These numerous required calculations are the reason that each point is not constrained to be non-negative immediately.

Thus, the function to be minimized is:

$$F = \sum_{i=1}^{n} (A + BX_i + CX_i^2 + DX_i^3 - y_i)^2 + \sum_{j=1}^{p} \lambda_j (B + 2CX_j + 3DX_j^2 - U_j)$$

where p is the number of points that were associated with a negative slope.

Taking the derivative of the function in respect to A, B, C, D, and λ (taking in respect to U is not beneficial) respectively, and setting them equal to zero gives the following set of equations.

$$nA + B\Sigma X_i + C\Sigma X_i^2 + D\Sigma X_i^3 = \Sigma Y_i$$

$$A\Sigma X_i + B\Sigma X_i^2 + C\Sigma X_i^3 + D\Sigma X_i^4 + \Sigma\lambda_j = \Sigma X_i Y_i$$

$$A\Sigma X_i^2 + B\Sigma X_i^3 + C\Sigma X_i^4 + D\Sigma X_i^5 + \Sigma 2\lambda_j X_j = \Sigma X_i^2 Y_i$$

$$A\Sigma X_i^3 + B\Sigma X_i^4 + C\Sigma X_i^5 + D\Sigma X_i^6 + \Sigma 3\lambda_j X_j = \Sigma X_i^3 Y_i$$

$$B + C2X_j + D3X_j^2 - U_j = 0$$

For $j = 1, 2, \ldots, p$

After solving for every combination of λ_j and U_j being set to zero, the solution for A, B, C, & D giving the minimum sum of squares is chosen.

After this, another check must be made to insure that this combination of coefficients does not allow the slope to be negative in the region of interest. If the slope is negative at any point, this point(s) must be added to the constraints and the operations repeated.

STATISTICAL SEPARATION OF

VARIABLE AND NON-VARIABLE COSTS IN THE

GEMINI SPACECRAFT PROGRAM

by Glen Self

Introduction

A major task during the last phase of this research grant has been to determine statistically oriented methodology which would provide a separation of the variable and non-variable or recurring and non-recurring costs associated with the subsystems of the Gemini spacecraft. Due to previous experience under other cost research contracts NASA/MSC preferred that a primarily statistical approach, which would be relatively insensitive to any assumptions, be made by the analysis performed under this phase of the grant. Due to the large and relatively complete file of subsystem cost data which was made available through NASA/MSC to the researchers on this grant, it was possible to perform analyses of the data which had previously been abandoned due to the lack of reasonable and consistent data. Through the efforts of Mr. Aubin Ferguson in ASDT/MSC it was possible to compile a data bank for this analysis. In order to demonstrate the methodology being developed by Texas A&M University in this area of cost segregation, a single subsystem, subsystem Number 37, the reactant supply subsystem, was chosen on a more or less random basis for purposes of demonstrating the techniques of methodology developed herein. The basic approach is to maximize the correlation of the various types of hardware deliveries with the cost categories available. This relationship is maximized through a simple correlation routine which will be described in more detail in those sections which follow. After this correlation routine has been used to establish the appropriate statistical lead-lag relationship within these data, the data are adjusted for lead-lag relationships in order that standard multivariate regression analysis could be performed to provide a predictive type model. In order to achieve even better and more realistic results the use of a convex programming technique was employed to determine both maximum and minimum of those costs which could be called variable during the program. A development of these techniques along with illustrative examples will be presented in the text which follows.

Establishing Lead-Lag Relationships Within the Hardware Delivery vs. Cost Data Picture

The basic philosophy of this phase of research was to relate physical hardware deliveries to those cost data which had been collected. One of the immediately obvious requirements upon inspection of the two groups of data was that there was a lead-lag relationship that appeared to exist between the hardware and the cost where both were being represented as discrete functions over time. In order to test the theories summarized above in relationship to variable and non-variable costs associated with spacecraft subsystem development and production, subsystem 37, reactant supply subsystem was selected primarily due to the fact that it was produced by one major sub-contractor and that the hardware delivery data was in a relatively useable format. In this particular subsystem there were ten different type of hardware deliveries plus a total accumulation of deliveries. These deliveries could be pin-pointed in time by delivery dates to the prime contractor, McDonald Aircraft Corp. The type of subsystems being considered were oxygen subsystem, the hydrogen subsystem, the dual pressure regulator, the hydrogen transducer, the oxygen transducer, the hydrogen pressure relief valve, the oxygen pressure relief valve, the low-pressure dual valve and the oxygen and hydrogen sybsystems were broken down into both long and short missions with the missions 5 and 7 being the long duration missions and missions 6, 8, 9, 10, 11 and 12 being the short duration missions in the Gemini program.

The non-zero cost available to this part of the study include the following: engineering, manufacturing, quality control, tooling, administrative cost by the prime contractor for sub-contractor programs, material and minor sub-contracts, ground support equipment, spares and major sub-contractor costs. These data were analyzed first at the major sub-contractor level. This was primarily due to the data being available on a monthly basis as opposed to those data being available for the major sub-contractors on a bi-annual basis. It was felt that if a statistically significant correlation was to be determined among deliveries and costs that the more detailed data would provide the better chance of establishing correlation patterns between the two data groups. Even though quarterly data were not used in this particular phase of the study, one advantage to its use in future studies might be due to the fact that it would have an averaging effect upon the bookkeeping being conducted on these hardware programs. The advantage to the averaging effect would be the elimination of some rather systematic variation which tends to indicate that the accounting records are adjusted toward the end of the year in order to more properly relect the total costs expended. The NASA data collection has tended to eliminate much of these over-estimation tendencies on the part of the prime and sub-contractors; however, quarterly grouping of the data might still further reduce variations of this nature. The cautions still exist that gross groupings would tend to eliminate the sensitivity of the data to the analysis technique being discussed in this section of the report. It should be pointed out that during the routine compilation of these data that were used in the analysis from various sources obvious inconsistencies were discovered. Based upon the knowledge of the mechanism of production and procurement, these were adjusted in a rational manner whenever possible. Generally, the first cost data point in each category was eliminated because it appeared to be an accumulation of some six to twelve months of cost being reported for the first time in the official bookkeeping setup for the cost reporting system on the form 533 for the prime contractor, McDonald. In order to avoid discussion of the specific data, the numerical values associated with costs and deliveries will be submitted under separate cover to NASA/MSC/ASTD, but will be referred to in this report with correlation and regression coefficients provided the reader.

From the standpoint of hardware deliveries and the fact that primarily the McDonald effort was being reviewed, another factor in the hardware delivery picture seem to be most significant and relevant to the cost analysis being performed. This was the existence of a number of rework items being returned to the vendor. For example, there were a total of 110 parts delivered and of those 50 were pointed out as having been in rework status and having been returned to the subcontractor on specific dates during the program. Due to McDonald's involvement in the return of hardware items to Airesearch, it was decided to include these events as part of the data analysis in an attempt to define the causal relationships for cost as nearly as possible.

In order to make the derivation of the recurring and non-recurring cost as statistically oriented as possible, the lead-lag relationships were analyzed using a correlation type analysis. This correlation was made between the observed delivery dates and the observed cost data. Since the cost data extended from August of 1962 up through November of 1966 and since delivery of hardware began in December of 1964 and terminated in August 1966, it was not obvious to the casual observer as to what the appropriate lead-lag relationship should be between the two groups of data. Therefore, a simple computation as shown in Formula 1 below

$$r_j = \frac{\sum_{i=1}^{n} \$_{ij} d_i - \sum_{i=1}^{n} \$_{ij} \sum_{i=1}^{n} d_i}{\Gamma_{\$_j} \Gamma_{d_i}} \qquad \begin{array}{l} i = 1, 2, \ldots n \\ (1) j = 1, \ldots, n-n+1 \end{array}$$

where j is an index of the starting point in the cost data series. This provided the analyst with a correlogram type analysis of the patterns which might exist in the data. Figure 1 displays the correlogram analysis for engineering cost vs. total deliveries. The particular type of pattern that is desireable within this analysis is one which has a relatively large positive correlation coefficient with much smaller values of correlation on either side. For example, if a lag of 10 months had a correlation of .9 with the lag at 9 and 11 months having a correlation near zero or negative, then it could be assumed that the correlation analysis had discovered a significant relationship between the two patterns observed in the cost data and the number of hardware deliveries on a time period by time period basis. The interpretation of these correlograms can be related to some degree to time series analysis of auto regressive data. That is, as the correlation coefficients cycle from large positive values to large negative values and back again, this tends to indicate a phasing-in and phasing-out of the agreement of the two patterns within the time periods being observed. Therefore, it can be seen that the desireable pattern described above would indicate a relatively good "fit". By using this technique to establish lead-lag relationships it is possible to avoid the introduction of subjective

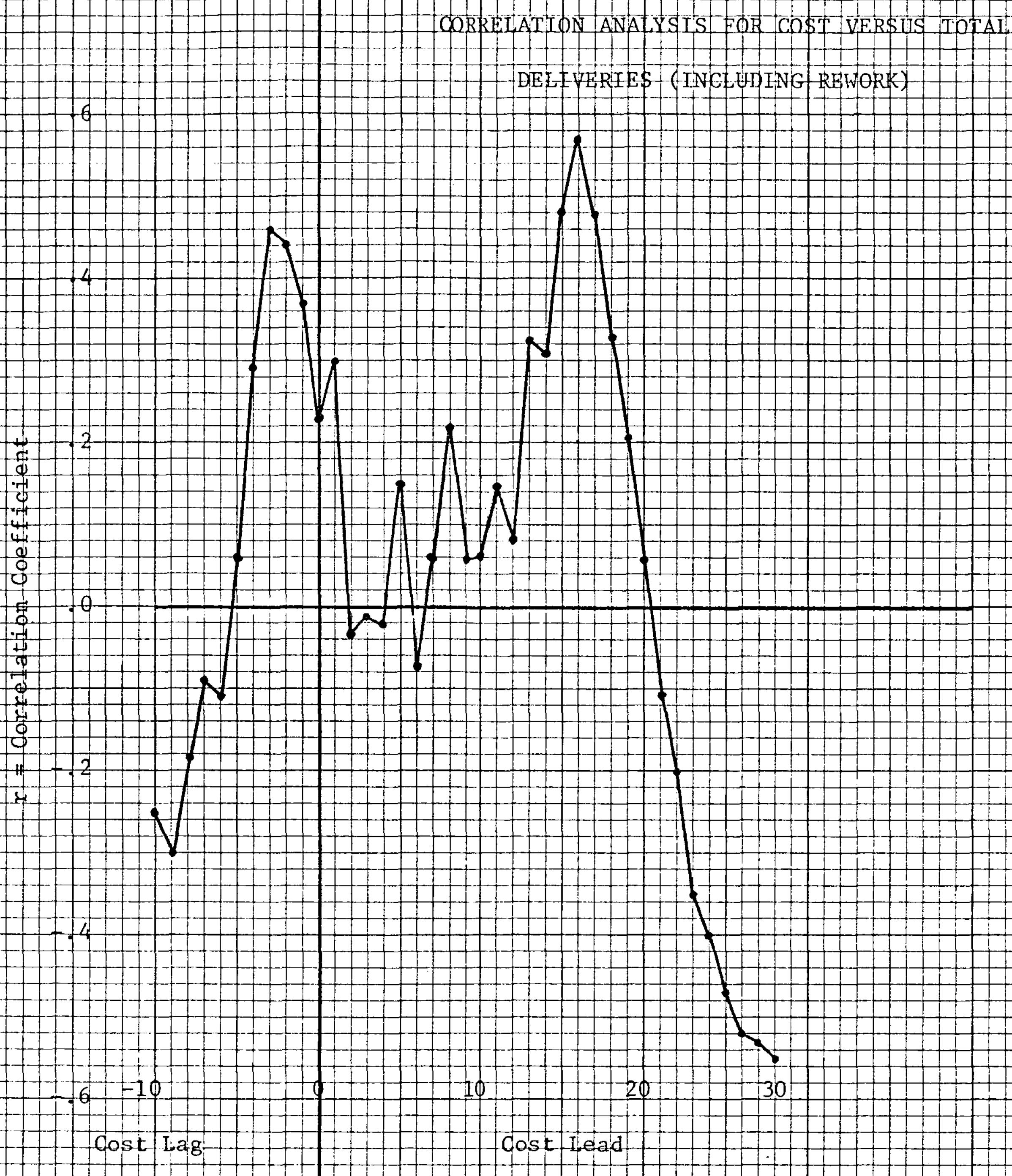
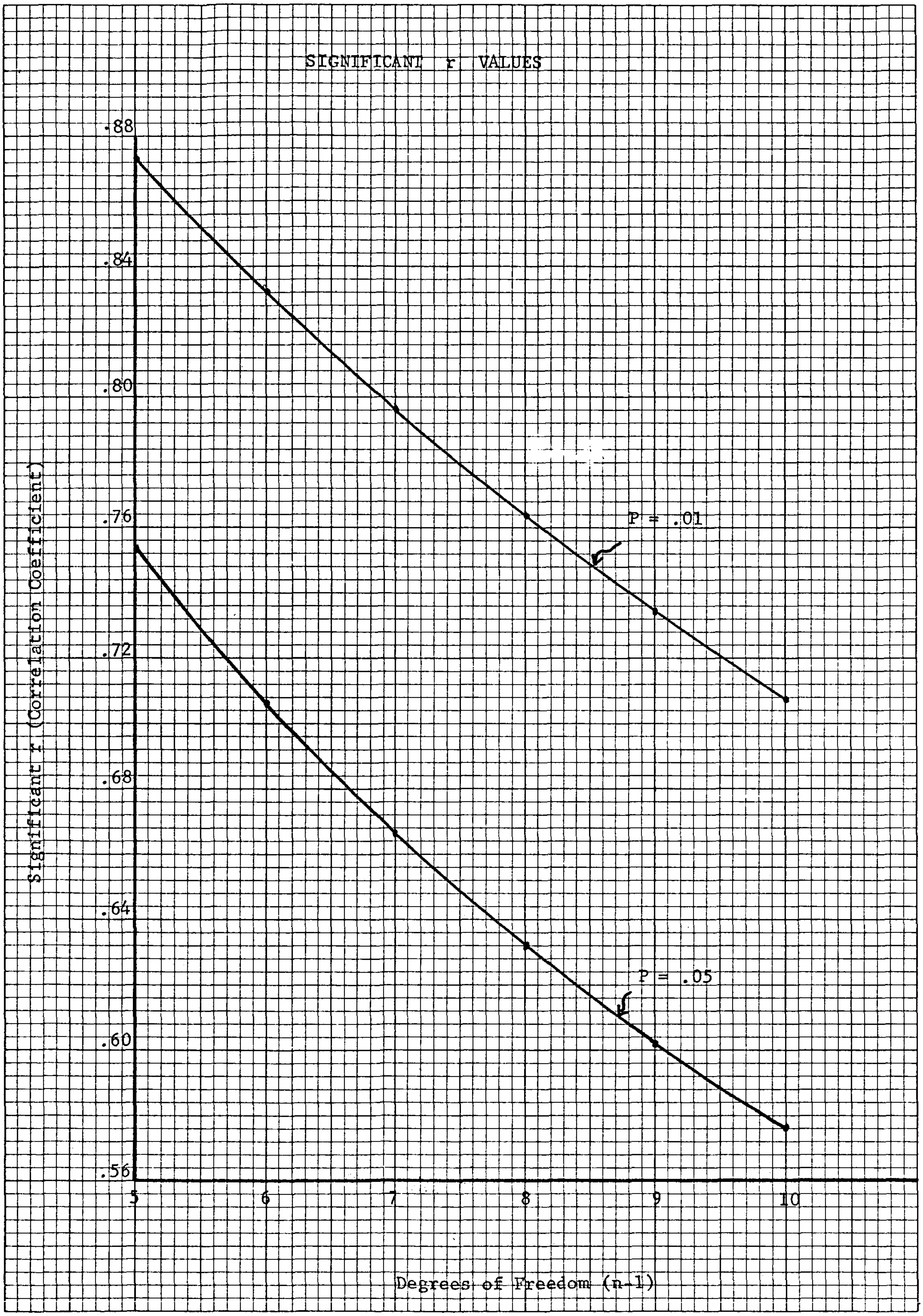


Figure 1. Delivery Versus Cost Lead-Lag Analysis For First Month of Cost Versus First Month In Which a Delivery Occurred.

estimates as to what these relationships might be. There is one small statistical danger in this type of approach, that is, with a limited number of deliveries the correlations might be expected to behave irrationally and perhaps give some false indication of correlation where, in fact, none did exist. In order to avoid this type of occurrence, Figure 2 has been included in this report for reference in case individuals choose to perform their own analysis of these types of data. A complete correlation analysis of all hardware types vs. all categories was performed on the data for Subsystem 37. In general, these analyses gave favorable results such that the lead-lag relationships could be established directly and the analysis of cost continued.

Computation of Variable Costs Through Constrained Regression

The continuation of the cost analysis utilized simple multiple linear regression techniques without weighting of the data except as that imposed by the restricting the constant term to be zero and requiring all coefficients to be positive in the equations. This is a legitimate mathematical programming approach to model building and does not permit the subjective mode to be introduced in the real sense. Initially, simple linear regression models were used to analyze the data. The results of this analysis was relatively good in that the explained variation of the regression analysis for all types of cost approached .8, that is to say, 80% or better of the variation observed in the cost data could be attributed to the hardware delivery variables which were included in the analysis as independent variables. By introducing time as an independent variable in the same simple linear regression case, it is possible to obtain an explained variation in excess of 90% in some of the cost categories which is a general indication of the time dependency of cost associated with these types of programs. Through the use of those techniques already developed,


SIGNIFICANT r VALUES
Significant r (Correlation Coefficient)
.88
.84
.80
.76
.72
.68
.64
.60
.56
P = .01
P = .05
5
6
7
8
9
10
Degrees of Freedom (n-1)

it has been possible to establish relatively good separation of variable and non-variable costs which do not appear to have very high sensitivity to any of the basic assumptions used in the analysis; however, by one more additional constraint of a physical type which will permit the determination of a maximizing or minimizing function with respect to non-variable costs over time, it will be possible to even more completely define the separation of non-variable and variable costs.

The research reported in this section of the final report has indicated the value of detail cost collection data during the course of programs such as Gemini. It is made possible a rather thorough look at those data which were collected and indicated types of data which should be collected on on-going and future spacecraft programs in order to continuingly up-grade cost estimation capability in both the long term subsystem and component level cost prediction techniques. It is visualized that a detail segregation of variable and non-variable costs will permit more exacting administration of extensions on production contracts and in the hardware procurement effort in general. The approach presented in this report has been one of achieving the best prediction techniques possible, the extension of the use of these results have been indicative as a part of the research effort; therefore, the application of the methodology derived by this research effort is left to the reader.

```
$JOB   961262414T4  2  1000      GLEN SELF CORRELATION ANALYSIS

$IBBOX          01-F
$EXECUTE        AGGIE
$IBFTC YYYYYY
      DIMENSION X(100), Y(100),R(100), LAG(100)
      READ(5,50) M,N
C     M = NUMBER OF COST DATA POINTS,  N = NUMBER OF DELIVERY DATA POINTS
   50 FORMAT(20X,2I3)
      DO 125 I=1,M
  125 READ (5,100) X(I),Y(I)
C     X(I) = MONTHLY COST DATA,  Y(I) = MONTHLY DELIVERY DATA
  100 FORMAT (2F5.0)
      FN = N
      LAG(1) = M-N
      J = 0.0
      K = 0.0
      WRITE (6,103)
  103 FORMAT (1H0, 10X, 24HCORRELATION COEFFICIENTS)
  106 SUMX = 0.0
      SUMY = 0.0
      SUMXY = 0.0
      SUMSQX = 0.0
      SUMSQY = 0.0
      NM = M-3
      DO 102 I=1,N
      SUMX = SUMX + X(I)
      SUMY = SUMY + Y(I)
      SUMXY = SUMXY + (X(I)*Y(I))
      SUMSQX = SUMSQX + (X(I)*X(I))
      SUMSQY = SUMSQY + (Y(I)*Y(I))
  102 CONTINUE
      SQ1 = (SUMX*SUMY)/FN
      RNUM = SUMXY - SQ1
      DEN1 = SUMSQX - ((SUMX*SUMX)/FN)
      DEN2 = SUMSQY - ((SUMY*SUMY)/FN)
      DEN3 = DEN1*DEN2
      RDEN = SQRT(DEN3)
      J=J+1
      K = K+1
      L = M-K
      R(J) = RNUM/RDEN
      WRITE (6,104) R(J), LAG(J)
      LAG(J+1) = LAG(J) - 1
  104 FORMAT (1H0, 22X, F10.6, 10X, 5HLAG = ,I5)
      IF (LAG(J+1)) 200,201,201
  200 FN = N+LAG(J+1)
  201 DO 189 I=1,M
      TEMP = X(1)
      X(I) = X(I+1)
      X(M) = 0.0
      Y(L) = 0.0
  189 CONTINUE
      IF(J-NM) 106,106,105
  105 CONTINUE
      STOP
      END
```

LIST OF REPORTS PREPARED UNDER
NASA GRANT SC-NGR-44-001-027

(Determination of Cost Curves)

1. Barnes, Wm. M., Production Cost Models, Texas A&M University, 1966

2. Britian, J. C., Computerized Evaluation of Cost Estimating Relationships, Texas A&M University, 1966.

3. Brown, S. P., Prediction of Run-out Costs, TA&MU, 1966

4. Brown, S. P., Program for Estimation of Run-out Costs TA&MU, 1966

5. Cooke, Wm. P., Research on Convex Programming Applied to Responce Surface Analysis and Related Problems of Statistical Reference, (Unpublished) TA&MU, 1965

6. Cooke, Wm. P., Constrained Estimation in Nonlinear Models; A Mathematical Programming Approach, TA&MU, 1966.

7. Cooke, Wm. P., Development of Cost Estimating Relationships to be used in the Spacecraft Cost Model, TA&MU, 1966.

8. Hartley, Dr. H. O., Hocking, Dr. R. R. and Cooke, Wm. P., Least Squares Fit of Definite Quadratic Forms by Convex Programming, TA&MU, 1966, (Submitted to Management Science, August, 1966)

9. Haynes, G. L., The Learning Curve, TA&MU, 1965

10. Haynes, G. L., Quantification and Utilization of Subjectively Determined Data in the Construction of Mathematical Models, TA&MU, 1966.

11. Haynes, G. L., Program to Facilitate Analysis of Expertise Data, TA&MU, 1966.

12. Self, Dr. G. D., Dynamic Programming Algorithm for Determining "Best Fit", TA&MU, 1966.

13. Self, Dr. G. D., Quantification of Subjectively Determined Data in the Formulation and Utilization of Mathematical Models, TA&MU, 1966.

14. Wortham, Dr. Wm. A., Servo Theory in Costing (Unpublished) TA&MU

15. Determination of Emperical & Analytical Spacecraft Parametric Curves - Theory & Methods, Progress Report I, SC-NGR-44-001-027, 1 December 1965, Industrial Engineering Department, TA&MU

16. Determination of Empirical & Analytical Spacecraft Parametric Curves-Theory & Methods, Progress Report I, SC-NGR-44-001-027, 1 May 1966, Industrial Engineering Department, TA&MU.

17. Determination of Empirical & Analytical Spacecraft Parametric Curves-Theory & Methods, Progress Report III, SC-NGR-44-001-027, 1 November 1966, Industrial Engineering Department, TA&MU.